INTERNATIONAL JOURNAL OF APPLIED ARTIFICIAL INTELLIGENCE AND ROBOTICS
E-ISSN: 3124-11212, DOI: 00.000/XXXX.YYYY.II.NNN
Vol. 2 No. 1 2026 pp. 49-60

Personalized Café Menu Recommendation Using Hybrid Collaborative and Content-Based Filtering Based on Location and User Interaction

Fika Adilah[1], Amirullah[2*]

[1,2] Department of Information Technology and Computers, Lhokseumawe State Polytechnic
Jl. Banda Aceh-Medan Km. 280 Buketrata 24301

Coressponding Author: amir@pnl.ac.id

Article info: Received 18-02-2026, Revised 02-03-2024, Accepted 18-03-2026

Abstract

Abstrack—The culinary industry, particularly cafes, has experienced rapid growth with the increasing number of cafes in various regions. Intense competition has driven café owners to innovate in attracting and retaining customers. A personalized menu recommendation system has become an effective solution to provide relevant services for each customer. This study employs a hybrid method that combines Collaborative Filtering and Content-Based Filtering to address this issue. Three cafes are included in this study: Station Coffee Premium in Kuta Blang, Ocean Coffee in Kampung Jawa Lama, and Bagi-Bagi Coffee in Lancang Garam. The research utilizes three main parameters: menu data, order history data, and café location data. By using these three parameters, the system will display menu recommendations at the cafes according to the customer's preferences. The research results indicate that the recommendation system performed well in black box testing, achieving a 100% success rate in all tested scenarios. Furthermore, method testing shows that the Content-Based Filtering method provides consistent results with stable precision, recall, and MAP across various scenarios. However, the Hybrid Filtering method proved to be the most accurate and relevant, combining the strengths of Content-Based Filtering and Collaborative Filtering to deliver menu recommendations that align with customer preferences based on their order history and location.

Keyword: recommendation, café, menu, relevant, hybrid

1. Introduction

The culinary industry, particularly cafés, is currently experiencing rapid growth alongside the increasing number of cafés emerging in various regions. The intensifying competition compels café owners to continually innovate and find ways to attract and retain customers. One strategy that can be implemented is providing personalized and relevant services for each customer [1]. An effective approach to realizing this strategy is the use of a personalized menu recommendation system [2].

The main challenge often faced by cafés is the difficulty in understanding the individual preferences of each customer. Without a deep understanding of what customers like and need, it becomes difficult for cafés to provide relevant and appealing menu recommendations. Most cafés still rely on standard menus that are the same for all customers, resulting in a less-than-optimal customer experience and limiting opportunities to increase sales and customer loyalty [3].

To address this issue, this study employs a hybrid method that combines Collaborative Filtering and Content-Based Filtering. Collaborative Filtering is a method that utilizes interaction data and preferences from a group of users to recommend items that might be liked by other users with similar patterns. This method is divided into two types: User-Based and Item-Based. User-Based Collaborative Filtering recommends items based on the similarity of preferences among users, while Item-Based Collaborative Filtering recommends items based on the similarity of characteristics among items [4].

Content-Based Filtering, on the other hand, recommends items by analyzing the characteristics of the items themselves. This method utilizes information such as item descriptions, categories, or other features to determine suitability with user preferences. By combining these two methods, the recommendation system is expected to provide more accurate and relevant recommendations. In addition, the integration of location information and user interaction enhances the level of personalization [5].

The implementation of this hybrid recommendation system in cafés offers several benefits, including improved customer experience, menu optimization, marketing efficiency, and better inventory management. With more relevant recommendations, customers will feel more valued and satisfied with personalized services, thereby increasing their loyalty to the café. Cafés can also optimize their offered menus by displaying items that appeal more to specific customers, thereby increasing the likelihood of sales. Personalized recommendations enable cafés to carry out more targeted and effective marketing. Furthermore, by understanding customer preferences, cafés can manage food ingredient inventory more efficiently and reduce waste [6]..

2. Research Methodology

This research involves the process of searching for information relevant to the topic, followed by collecting the gathered data for further analysis.

2.1. System Requirements Analysis

The system analysis includes functional requirements analysis and non-functional requirements analysis.

2.2. Functional Requirements

Functional requirements include several needs for each entity that detail the steps carried out to facilitate the processes within the system.

1. Functional Requirements for Customers.
   a. Customers can register an account.
   b. ii. Customers can log in by entering their login information.
   c. iii. Customers can view and select menus from various cafés available in the system.
   d. iv. Customers can place orders.
   e. v. Customers can log out of the system
2. Functional Requirements for Café Owners (Owners).
   a. Owners can register and register the café name.
   b. ii. Owners can log in by entering the data registered previously.
   c. iii. Owners can create accounts for staff members, including waiters, kitchen staff, and cashiers.
   d. iv. Owners can update the café profile.
   e. v. Owners can input café menu category data.
   f. vi. Owners can input café menu data.
   g. vii. Owners can input the number of available tables.
   h. viii. Owners can view café sales reports.
   i. ix. Owners can log out of the system.
3. Functional Requirements for Waiters.
   a. Waiters can log in by entering their login information.
   b. ii. Waiters can add orders from customers.
   c. iii. Waiters can receive orders from customers.
   d. iv. Waiters can log out of the system.
4. Functional Requirements for Kitchen Staff.
   a. Kitchen staff can log in by entering their login information.
   b. ii. Kitchen staff can view customer order information data.
   c. iii. Kitchen staff can process customer orders.
   d. iv. Kitchen staff can log out of the system.
5. Functional Requirements for Cashiers
   a. Cashiers can log in by entering their login information.
   b. ii. Cashiers can view customer order information data.
   c. iii. Cashiers can process payment transactions from customers.
   d. iv. Cashiers can log out of the system.

2.3. Non-Functional Requirements

Non-functional requirements include several needs related to hardware and software. Hardware refers to the physical components of a computer that can be seen and touched, while software is a set of instructions executed by the computer to complete specific tasks. The following are some non-functional requirements of this system.

1. Hardware

In the development of this thesis and system, a single laptop was used to complete the entire process of system creation and research reporting, specifically using a Lenovo laptop

2. Software

The following are the software specifications used in the system development, as shown in Table I below:

table 1 Software Specifications

No	Name	Specification	Description
1	Operating System	Windows 11	Server Operating System
2	Google Chrome	Version 126.0.6478.183 (Official Build) (64-bit)	To Access the System Page
3	Visual Studio Code	Version 1.91.1	code editor
4	Draw.io	Online	Used for system design creation
5	Postman	Version 11.5.1	Used for API testing
6	Laragon	Version 8.3.30	Local web server on the computer
7	Figma	Version C6	Used for interface design creation
8	ERDPlus	Online	Used for creating ERD (Entity Relationship Diagram)
9	Laravel	Version 11x	Framework for backend development
10	Vue.js	Vue 3	Framework for frontend development
11	MySQL	MySQL 8.0	Database for data storage
12	Mailtrap	Online	Used for email verification testing

2.4. System Design (software / hardware)

System design begins with creating a use case diagram for each activity performed by each user interacting with the system. The main purpose of designing a use case diagram is to illustrate all the functions contained within the system and how each of these functions is used by its users.

Figure 1 Use case diagram

2.5. Research Method Design

The design of the hybrid filtering method uses two research approaches by combining the Content-Based Filtering and Collaborative Filtering methods, as shown in Figures 2, 3, and 4.

Figure 2 Content-Based Filtering Design

Figure 2 above illustrates the stages of the filtering process using Content-Based Filtering. The process begins by collecting user order data and location. Then, the system checks whether the user has placed an order before. If yes, the system identifies the categories of menus previously ordered and searches for menus based on the user's location and the similarity of those categories. If not, the system uses the user's location and all available categories to provide recommendations. Finally, the system displays the recommended menus to the user, and the process is completed.

Figure 3 Collaborative Filtering Design

Figure 3 above explains the stages of the filtering process using Collaborative Filtering. The process begins by collecting order data from all users, then sorting the menus based on the number of orders. After that, the system selects the top 10 menus and considers the user's location to provide recommendations. Finally, the system displays the recommended menus to the user, and the process is completed.

Figure 4 Hybrid Filtering Design

Figure 4 above explains the stages of combining the two methods, CBF and CF. The process begins by merging the recommendation results from both methods. After that, duplicates in the recommendation list are removed to ensure each item in the list is unique. The final step is to display the final recommendations, which are free from duplicates, to the user.

2.6. Testing Technique Design

In designing a system, testing is necessary to evaluate the quality and functionality of the system. In this case, two testing techniques are used: Black Box Testing and White Box Testing.

3. Result and Discussions

This results and discussion section will review the implementation outcomes of the hybrid method that combines Content-Based Filtering and Collaborative Filtering, as well as discuss the results of system testing.

3.1. System Display Results

The system display results refer to the appearance of the login page, where users are required to enter their email address and password in the provided fields. After that, users can click the "Sign In" button to proceed to the homepage.

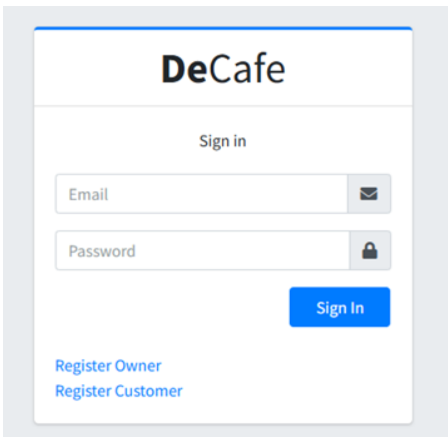


Figure 5 Customer Homepage

3.2. Black Box Testing

This test will evaluate the login page, as shown in Table 2.

Table 2 Login Page Testing

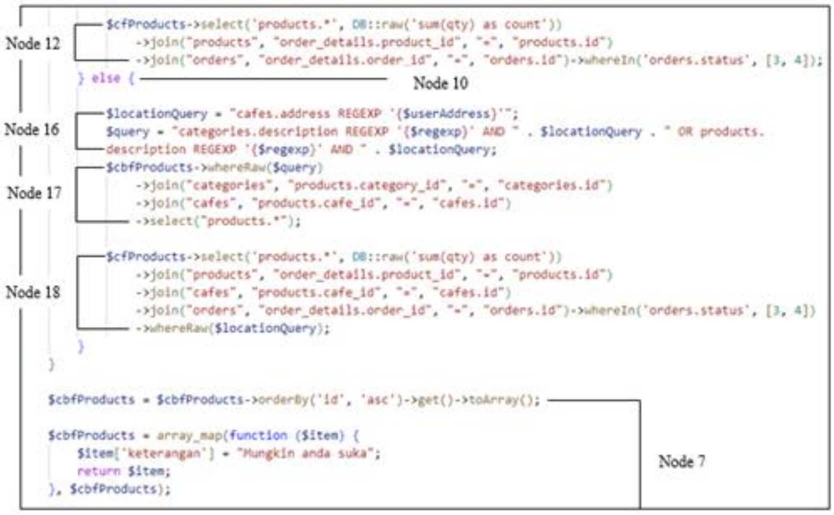
No	Test Case	Pre-Condition	Test Steps	Test Data	Expected Result	Post Condition	Actual Result	Status
1	Login	User enters valid data	1. Fill in Email 2. Fill in Password 3. Click the Sign button	User Account	Login successful and redirects to the dashboard page	Login successful and displays the dashboard page	Login successful and displays the dashboard page	PASS

3.3. White Box Testing

This test will evaluate the recommendation system code used as data in this testing process.

Figure 6 Recommendation System Code Lines

Figure 7 Recommendation System Code Lines



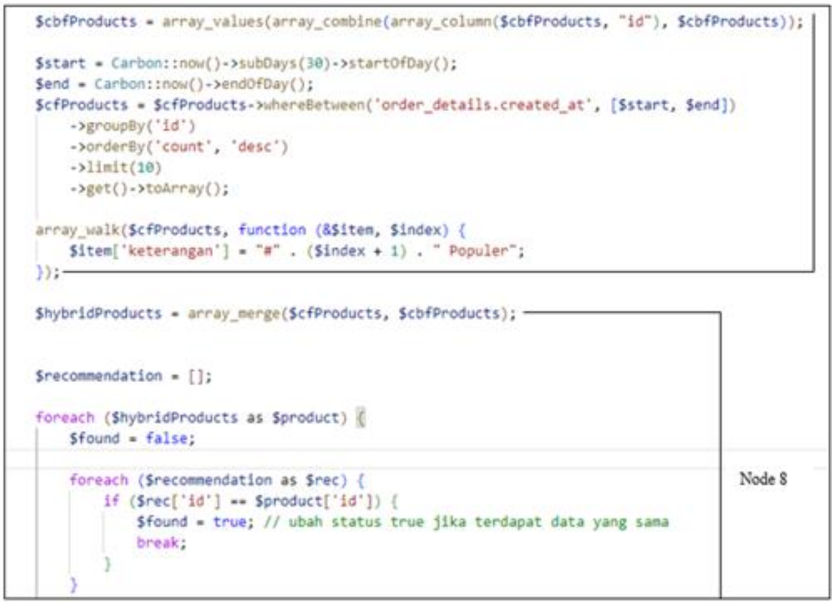

```
        $cfProducts->select('products.*', DB::raw('sum(qty) as count'))
            ->join("products", "order_details.product_id", "=", "products.id")
            ->join("orders", "order_details.order_id", "=", "orders.id")->whereIn('orders.status', [3, 4]);
    } else {
        $locationQuery = "cafes.address REGEXP '{$userAddress}'";
        $query = "categories.description REGEXP '{$regexp}' AND " . $locationQuery . " OR products.
        description REGEXP '{$regexp}' AND " . $locationQuery;
        $cbfProducts->whereRaw($query)
            ->join("categories", "products.category_id", "=", "categories.id")
            ->join("cafes", "products.cafe_id", "=", "cafes.id")
            ->select("products.*");

        $cfProducts->select('products.*', DB::raw('sum(qty) as count'))
            ->join("products", "order_details.product_id", "=", "products.id")
            ->join("cafes", "products.cafe_id", "=", "cafes.id")
            ->join("orders", "order_details.order_id", "=", "orders.id")->whereIn('orders.status', [3, 4])
            ->whereRaw($locationQuery);
    }
}

$cbfProducts = $cbfProducts->orderBy('id', 'asc')->get()->toArray();

$cbfProducts = array_map(function ($item) {
    $item['keterangan'] = "Mungkin anda suka";
    return $item;
}, $cbfProducts);
```

Figure 8 Recommendation System Code Lines




```
$cbfProducts = array_values(array_combine(array_column($cbfProducts, "id"), $cbfProducts));

$start = Carbon::now()->subDays(30)->startOfDay();
$end = Carbon::now()->endOfDay();
$cfProducts = $cfProducts->whereBetween('order_details.created_at', [$start, $end])
    ->groupBy('id')
    ->orderBy('count', 'desc')
    ->limit(10)
    ->get()->toArray();

array_walk($cfProducts, function (&$item, $index) {
    $item['keterangan'] = "#" . ($index + 1) . " Populer";
});

$hybridProducts = array_merge($cfProducts, $cbfProducts);


$recommendation = [];

foreach ($hybridProducts as $product) {
    $found = false;

    foreach ($recommendation as $rec) {
        if ($rec['id'] == $product['id']) {
            $found = true; // ubah status true jika terdapat data yang sama
            break;
        }
    }
```

Figure 9 Recommendation System Code Lines

As we can see in Figures 6, 7, 8, and 9, these are the lines of code used to process the recommendation system, which will then be detailed in the form of a scenario flowchart as shown in the figure below.

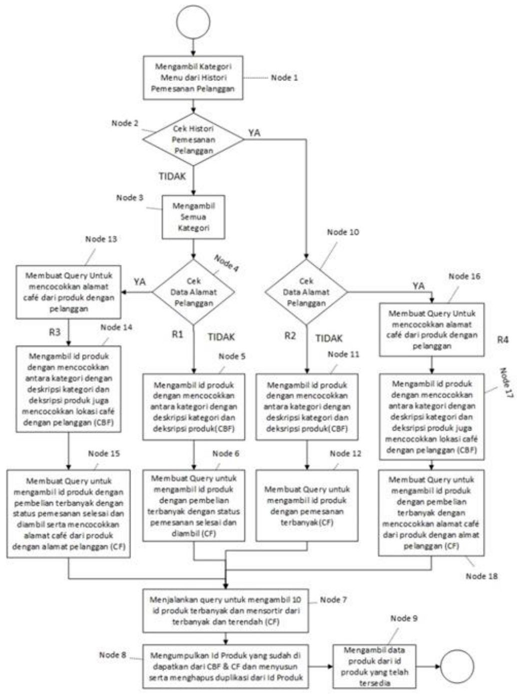


Figure 10 Scenario Flowchart

Based on the flowchart in Figure 6, it is known that the number of edges (E) = 20, which are the lines connecting nodes; the number of nodes (N) = 18, which represent activities; the number of predicates (P) = 3, which are branching nodes; and the number of regions indicating areas in the flowchart, which can be seen in symbols R1 to R4 in the figure above. Thus, if we calculate using the cyclomatic complexity formula in Table III and the Graph Matrix in Table IV, the results are as follows.

Table 3 Cyclomatic Complexity Of The Recommendation System

V(G) = E – N +2	V(G) = P + 1	V(G) = R
V(G) = 20 – 18 +2	V(G) = 3 + 1	V(G) = 4
V(G) = 4	V(G) = 4	

table 4 Graph Matrix Table

X	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	Hasil
1		1																	0
2			1							1									1
3				1															0
4					1								1						1
5						1													0
6							1												0
7								1											0
8									1										0
9																			0
10											1					1			1
11												1							0
12							1												0
13														1					0
14															1				0
15							1												0
16																	1		0
17																		1	0
18							1												0
Jumlah																			4

Based on the cyclomatic complexity calculation in Table III, the number of independent paths for the recommendation system route is 4, with the independent paths as follows:

Path 1: 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9
Path 2: 1 - 2 - 10 - 11 - 12 - 7 - 8 - 9
Path 3: 1 - 2 - 3 - 4 - 13 - 14 - 15 - 7 - 8 - 9
Path 4: 1 - 2 - 10 - 16 - 17 - 18 - 7 - 8 – 9

After determining the independent paths, it is known that the data provisioning flow of the recommendation system consists of 4 types of data routes. These 4 data route paths have a low level of risk with straightforward procedures. After determining the 4 data provisioning paths (independent paths), the next step is to conduct testing.

table 5 White Box Testing Table

No	Rute IP	Description	Expected Result	Actual Result	Pass / Fail
1	1-2-3-4-5-6-7-8-9	If there is no order history and no address has been added.	Displays default data by matching all category names to product descriptions and products with the highest number of completed and collected orders.	Successfully displays default data by matching all category names to product descriptions and products with the highest number of completed and collected orders.	PASS
2	1-2-10-11-12-7-8-9	If there is an existing order history and no address has been added.	Displays product data by matching the categories of previously ordered products and the most frequently ordered products.	Successfully displays product data by matching the categories of previously ordered products and the most frequently ordered products.	PASS
3	1-2-3-4-13-14-15-7-8-9	If there is no order history and an address has been added.	isplays default data by matching all category names to product descriptions, as well as matching the café address from the products with the customer's address and products with the highest number of completed and collected orders, along with café address matching between the products and the customer's address.	Successfully displays default data by matching all category names to product descriptions, as well as matching the café address from the products with the customer's address and products with the highest number of completed and collected orders, along with café address matching between the products and the customer's address.	PASS
4	1-2-10-16-17-18-7-8-9	If there is an existing order history and an address has been added.	Displays product data by matching the categories of previously ordered products, as well as matching the café address from the products with the customer's address and the most frequently ordered products, along with café address matching between the products and the customer's address.	Successfully displays product data by matching the categories of previously ordered products, as well as matching the café address from the products with the customer's address and the most frequently ordered products, along with café address matching between the products and the customer's address.	PASS

After determining the four data provisioning paths, it can be concluded that the success rate from the four obtained paths is 100%, and the failure rate is 0%. Next, an analysis of the recommendation system testing is carried out using several calculations based on evaluation metric formulas such as Precision, Recall, and Mean Average Precision (MAP):

Precision

Precision measures the proportion of items recommended by the system that are relevant to the user. Precision is calculated using the following formula:

$$Precision = \frac{Number\ of\ Relevant\ Products\ Recommended}{Total\ Number\ of\ Products\ Recommended}$$

Recall

Recall measures how well the system identifies all items that are relevant to the user. Recall is calculated using the following formula:

$$Precision = \frac{Number\ of\ Relevant\ Products\ Recommended}{Total\ Number\ of\ Relevant\ Products}$$

The method testing will also be conducted using the four existing paths to evaluate how well the recommendations perform in each path by comparing Content-Based Filtering, Collaborative Filtering, and Hybrid Filtering.

The user has no order history and has not added an address.

The results of the method testing for this user are as follows:

table 6 User Has No Order History And Has Not Added An Address

Method	*Precision*	*Recall*
Content-Based Filtering (CBF)	$\frac{36}{100} = 0.36\ (36\%)$	$\frac{36}{100} = 0.36\ (36\%)$
Collaborative Filtering (CF)	$\frac{5}{10} = 0.5\ (50\%)$	$\frac{5}{100} = 0.05\ (5\%)$
Hybrid Filtering (HF)	$\frac{36}{100} = 0.36\ (36\%)$	$\frac{36}{100} = 0.36\ (36\%)$

The user has an order history and has not added an address.

The results of the method testing for this user are as follows:

table 7 User Has An Order History And Has Not Added An Address

Method	Precision	Recall
Content-Based Filtering (CBF)	$\frac{36}{100} = 0.36\ (36\%)$	$\frac{36}{100} = 0.36\ (36\%)$
Collaborative Filtering (CF)	$\frac{5}{10} = 0.5\ (50\%)$	$\frac{5}{100} = 0.05\ (5\%)$
Hybrid Filtering (HF)	$\frac{36}{100} = 0.36\ (36\%)$	$\frac{36}{100} = 0.36\ (36\%)$

The user has no order history and has added an address.

The results of the method testing for this user are as follows:

table 8 User Has No Order History And Has Added An Address

Method	Precision	Recall
Content-Based Filtering (CBF)	$\frac{12}{35} = 0.34\ (34\%)$	$\frac{12}{35} = 0.34\ (34\%)$
Collaborative Filtering (CF)	$\frac{4}{4} = 1\ (100\%)$	$\frac{4}{35} = 0.11\ (11\%)$
Hybrid Filtering (HF)	$\frac{12}{35} = 0.34\ (34\%)$	$\frac{12}{35} = 0.34\ (34\%)$

The user has an order history and has added an address.

The results of the method testing for this user are as follows:

table 9 User Has An Order History And Has Added An Address

Method	Precision	Recall
Content-Based Filtering (CBF)	$\frac{11}{20} = 0.55\ (55\%)$	$\frac{11}{20} = 0.55\ (55\%)$

Collaborative Filtering (CF)	$\frac{4}{6} = 0.66\ (66\%)$	$\frac{4}{20} = 0.2\ (20\%)$
Hybrid Filtering (HF)	$\frac{11}{20} = 0.55\ (55\%)$	$\frac{11}{20} = 0.55\ (55\%)$

Mean Average Precision (MAP)
MAP is used to evaluate how well the model recommends relevant items among the list of recommended items. The MAP value ranges from 0 to 1, where a higher value indicates that the model performs better in recommending relevant items at the top positions in the list.

$$\text{Mean Average Precision (MAP)} = \frac{1}{Q}\sum_{i=1}^{Q} AP_i$$

Before calculating the Mean Average Precision (MAP), we can first measure the Average Precision (AP) for each method from each path.

$$AP = \frac{1}{N}\sum_{k=1}^{N} P(k) \times rel(k)$$

table 10 Average Precision (Ap)

Method	*Average Precision* (AP)
Content-Based Filtering (CBF)	$((0.36 \times 0.36) + (0.36 \times 0.36) + (0.34 \times 0.34) + (0.55 \times 0.55))/4 = 0{,}67/4 = 0.16$
Collaborative Filtering (CF)	$((0.5 \times 0.05) + (0.5 \times 0.05) + (1 \times 0.11) + (0.66 \times 0.2))/4 = 0{,}29/4 = 0.07$
Hybrid Filtering (HF)	$((0.36 \times 0.36) + (0.36 \times 0.36) + (0.34 \times 0.34) + (0.55 \times 0.55))/4 = 0{,}67/4 = 0.16$

After determining the AP value for each method, the next step is to calculate the Mean Average Precision (MAP):

table 11 Mean Average Precision (Map)

Method	*Mean Average Precision* (MAP)
Content-Based Filtering (CBF)	$\frac{0.36 + 0.36 + 0.34 + 0.55}{4} = \frac{1.61}{4} = 0.4025\ (40.25\%)$
Collaborative Filtering (CF)	$\frac{0.5 + 0.5 + 1 + 0.66}{4} = \frac{2.66}{4} = 0.665\ (66.5\%)$
Hybrid Filtering (HF)	$\frac{0.36 + 0.36 + 0.34 + 0.55}{4} = \frac{1.61}{4} = 0.4025\ (40.25\%)$

From the test results, it can be seen that the Collaborative Filtering method produces higher Precision and MAP values compared to Content-Based Filtering, but with a lower Recall value. This indicates that Collaborative Filtering is more accurate in recommending items, while Content-Based Filtering performs better in covering all relevant items.

4. Conclusion

The conclusion obtained after conducting research on the menu recommendation system using the hybrid method—combining Content-Based Filtering and Collaborative Filtering—is as follows. Testing was carried out through two approaches: black box testing and method testing. The results of the black box testing showed that all test cases were successfully executed as expected. This indicates that the system functions properly in all given testing scenarios. Method testing was performed using evaluation metrics such as precision and recall to measure the effectiveness of recommendations. Content-Based Filtering (CBF) tended to produce consistent precision and recall values across different scenarios, ranging between 34%–55%. Collaborative Filtering (CF) showed more variable performance, with high precision in some cases (up to 66%) but relatively low recall (as low as 5% in certain scenarios). Hybrid Filtering (HF) provided consistent results with precision and recall values equal to or better than CBF, especially for users with existing order histories. Overall, the testing results indicate that the recommendation system using the hybrid method can deliver accurate and relevant results, with improved performance by combining the strengths of both methods used.

REFERENCES

[1] M. M. Sihombing, M. H. Arifin, and M. Maryono, "Pengaruh Varian Menu, Harga, dan Suasana Cafe, Terhadap Kepuasan Konsumen Cafe Miltie Garden Mulawarman Banjarmasin," Smart Bus. J., vol. 1, no. 1, p. 26, 2022, doi: 10.20527/sbj.v1i1.12787.

[2] F. Ricci, L. Rokach, and B. Shapira, Recommender Systems Handbook, no. October. 2011. doi: 10.1007/978-0-387-85820-3.

[3] T. D. Andini and A. Zulkarnain, "Suggestions Friends Engine Berbasis Hybrid Recommender System Untuk Mendapatkan Rekomendasi Teman Terbaik Pada Web Jejaring Sosial," J. Ilm. Teknol. Inf. Asia, vol. 7, no. 2, 2013.

[4] Y. I. Lubis, D. J. Napitupulu, and A. S. Dharma, "Implementasi Metode Hybrid Filtering (Collaborative dan Content-based) untuk Sistem Rekomendasi Pariwisata," Dep. Tek. Elektro dan Teknol. Informasi, FT UGM, pp. 28–35, 2020.

[5] R. H. Mondi and A. Wijayanto, "Recommendation System With Content-Based Filtering Method for Culinary Tourism in Mangan Application," ITSMART J. Ilm. Teknol. dan Inf., vol. 8, no. 2, pp. 65–72, 2019.

[6] X. Zhou and Y. Shi, "DeepFaFM :A Field-array Factorization Machine based Neural Network for CTR Prediction," Proc. 2020 IEEE 4th Inf. Technol. Networking, Electron. Autom. Control Conf. ITNEC 2020, pp. 2559–2563, 2020, doi: 10.1109/ITNEC48623.2020.9084654.

[7] D. Purwanto, "Rekomendasi Paket Pembelian Barang Pada Toko Online Dengan Collaborative Filtering," Semin. Nas. Sains dan Teknol. Terap. III, no. August, 2015, [Online]. Available: https://www.researchgate.net/profile/Devi-Purwanto/publication/319256110_REKOMENDASI_PAKET_PEMBELIAN_BARANG_PADA_TOKO_ONLINE_DENGAN_COLLABORATIVE_FILTERING/links/599e4b9945851574f4b3606b/REKOMENDASI-PAKET-PEMBELIAN-BARANG-PADA-TOKO-ONLINE-DENGAN-COLLABORA

[8] L. Tommy, D. Novianto, and Y. S. Japriadi, "Sistem Rekomendasi Hybrid untuk Pemesanan Hidangan Berdasarkan Karakteristik dan Rating Hidangan," J. Appl. Informatics Comput., vol. 4, no. 2, pp. 137–145, 2020, doi: 10.30871/jaic.v4i2.2687.

[9] F. Axza, F. Sofi'ie, and A. Qoiriah, "Analisis Perbandingan Framework Front-End Javascript React dan Vue Pada Pengembangan Website," J. Informatics Comput. Sci., vol. 05, pp. 157–164, 2023.

[10] R. Pangestika and R. T. Dirgahayu, "Pengembangan Back-end Sistem Informasi Pendataan Sekolah Desa Komunitas Pendar Foundation Yogyakarta," J. Pros. Autom., vol. 1, no. 2, pp. 1–6, 2020, [Online]. Available: https://journal.uii.ac.id/AUTOMATA/article/view/15548

[11] Jamaluddin et al., BUKU (Book Chapter)-Sistem Basis Data (Elmi Devia)_oke. 2022.

[12] J. Ben Schafer, D. Frankowski, J. Herlocker, and S. Sen, "LNCS 4321 - Collaborative Filtering Recommender Systems," no. January 2007, 2014.

[13] B. A. Priyaungga, D. B. Aji, M. Syahroni, N. T. S. Aji, and A. Saifudin, "Pengujian Black Box pada Aplikasi Perpustakaan Menggunakan Teknik Equivalence Partitions," J. Teknol. Sist. Inf. dan Apl., vol. 3, no. 3, p. 150, 2020, doi: 10.32493/jtsi.v3i3.5343.

[14] J. B. L. Sie, Izmy Alwiah Musdar, and Syamsul Bahri, "Pengujian White Box Testing Terhadap Website Room Menggunakan Teknik Basis Path," KHARISMA Tech, vol. 17, no. 2, pp. 45–57, 2022, doi: 10.55645/kharismatech.v17i2.235.